

# Der Editor VIM

(Vi IMproved)

# Der Editor VIM

(Vi IMproved)

- Der VIM (VI Improved) ist eine Weiterentwicklung des bekannten Texteditors VI, von Baam Monar als OpenSource Programm 1991 veröffentlicht.
- Er funktioniert im Textmodus auf **jedem Terminal**
- Die **Bedienung** erfolgt ausschließlich über die **Tastatur**
- Der VIM zeichnet sich durch seine verschiedenen **Betriebsmodi** aus.
- Er beherrscht:
  - **Syntaxhervorhebung**, Programmiersprachen abhängig
  - eigene **Skriptsprache** zur Funktions-Erweiterung
  - **multiple Puffer-** und Split-Screens
  - uvm.

# Schnellstart

- Aufruf des VIM:  
`wolfgang@nyx:~$ vim meineDatei.c`
- Wechsel in den *Einfüge-Modus* und Einfügen von Text:  
`a`
- Speichern des aktiven Dokumentes:  
`<ESC> :w`
- Beenden des VIM:  
`<ESC> :q`

# Modi des VIM

Der VIM unterscheidet drei verschiedene **Modi**. Der jeweils aktive Modus legt fest, wie das eingegebene Zeichen zu interpretieren ist.

- **Befehls-Modus (Command Mode)**  
Alle Eingaben werden als Kommandos betrachtet. Es erfolgt keine Einfügung am aktiven Dokument.  
Nach dem Start befindet sich der VIM in der Regel im Kommando-Modus
- **Einfüge-Modus (Insert Mode)**  
Die Eingaben werden in das aktive Dokument übernommen.  
Mit *ESC* wird der Einfüge-Modus wieder verlassen.
- **Kommandozeilen-Modus (Command-line Mode)**  
Er gestattet die Eingabe einzeliger Kommandos am Bildschirmrand  
Ein Kommando in diesem Modus beginnt mit ':', '/', '?' oder '!'.  
Nach Ausführung des Kommandos wird dieser Mode wieder verlassen

# Einige Kommandos

## Positionierung des Cursors

- **[n] G**  
Sprung zu einer bestimmten Zeile  
Beispiel: **25 G** für Zeile 25
- **[n] |**  
Sprung zu einer bestimmten Spalte
- **[n] w**  
Positioniert den Cursor n Wörter vorwärts
- **[n] b**  
Positioniert den Cursor n Wörter vorwärts
- **m Marke**  
Setzt eine Marke an die aktuelle Cursor-Position.  
Beispiel: **ma** setzt Marke a
- **' Marke**  
Sprung zu einer vorher gesetzten Marke

# Einige Kommandos

## Löschen und Einfügen von Zeichen

- **i**  
Einfügen vor dem aktuellen Zeichen
- **A**  
Einfügen am Zeilenende
- **x**  
Zeichen unter dem Cursor löschen
- **D**  
Löschen ab Cursor bis zum Ende der Zeile
- **[n]J**  
Verbinden der Zeile mit der Nachfolgenden Zeile
- **[n]dw**  
Löschen bis zum Anfang des nächsten Wortes

# Einige Kommandos

## Suchen und Ersetzen

- **/Muster** bzw. **?Muster**  
Suche des Musters nach vorne bzw. hinten
- **n** bzw. **p**  
Ist auf obige Suche anzuwenden und positioniert den Cursor vor dem nächsten bzw. dem vorherigen gefundenen Muster
- **:[Bereich]s /alt/neu/Modifikator**  
Ersetzen innerhalb eines Zeilenbereichs  
Der **Modifikator** bestimmt die Wirkungsweise der Ersetzung:
  - c: jede Ersetzung muss bestätigt werden
  - g: ersetzt alle Vorkommen im Dokument von alt
  - i : Groß- und Kleinschreibung ignoierenBeispiel:  
**: 1,3 s/dei/die/c**  
(Ersetzt in Zeile 1 und 3 alle Vorkommen von 'die' durch 'die', wobei jede Ersetzung bestätigt werden muss)

# Einige Kommandos

## Suchen und Ersetzen von Zeilen

- **[n]dd**  
Löschen (in den Puffer) von n Zeilen ab Cursorposition
- **[n]yy**  
n Zeilen in den Puffer kopieren
- **p**  
Einfügen des Puffers („letzte Löschung“) an der Cursorposition
- **o**  
Einfügen einer Leerzeile oberhalb der Cursorposition

-

# Einige Kommandos

## Sonstige nützliche Kommandos

- **:q** bzw. **:q!**  
Beenden bzw. Beenden ohne zu speichern des Editors
- **:w**  
Speichern des aktiven Dokuments
- **u**  
Undo – macht die letzte Aktion rückgängig
- **:sy on** bzw. **:sy off**  
Syntax Highlighting an- bzw. ausschalten
- **:set nu** bzw. **:set nonu**  
Zeilennummern Ein- bzw. Ausblenden
- **<ESC>**  
*Wechselt in den Kommando-Modus*

# VIM anpassen

## Die .vimrc

- Um den VIM individuell anzupassen, empfiehlt es sich die .vimrc zu editieren (Achtung: [PUNKT]vimrc)

```
eric@turing ~ $ vim .vimrc
```

- Hier sollten nun folgende Zeilen unten angefügt werden:

```
set nocompatible  
set backspace=indent,eol,start  
set autoindent  
set ruler  
set paste  
syntax on  
set showmatch
```